



Finding equidistant nondominated points for biobjective mixed integer programs

Kidd, Martin Philip; Lusby, Richard Martin ; Larsen, Jesper

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Kidd, M. P., Lusby, R. M., & Larsen, J. (2016). *Finding equidistant nondominated points for biobjective mixed integer programs*.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Finding equidistant nondominated points for biobjective mixed integer programs

Martin Philip Kidd, Richard Lusby & Jesper Larsen

December 7, 2016

Abstract

The nondominated frontier of a multiobjective optimization problem can be overwhelming to a decision maker, as it is often either exponential or infinite in size. Instead, a representation of this set in the form of a small sample of points is often preferred. In this paper we present a new biobjective criterion space search method for generating a small set of equidistant points based on the space division idea behind Voronoi diagrams. The motivation for this method stems from the finding that there exists a dual relationship between the well-established quality measures of coverage and uniformity, and that a set of equidistant points closes the gap. The method is easy to implement, and relies only on the availability of a black-box solver. We show on a benchmark set of biobjective mixed integer programming instances that the method outperforms the state of the art with respect to both coverage and uniformity.

1 Introduction

Multiobjective optimization is concerned with the problem of presenting a Decision Maker (DM) with a set of alternative feasible solutions to an optimization problem that represents a tradeoff between multiple objectives. It is natural to restrict the search to *efficient solutions*, i.e. finding a feasible solution that cannot be improved upon with respect to any objective without degrading its value with respect to another objective, and the problem of finding all possible efficient solutions has received much attention in the literature (Stidsen et al., 2014; Boland et al., 2015a,b, 2016a,b; Ehrgott et al., 2016). However, for practical problems the entire efficient set can easily become too large for a DM to manage, and so a more practical goal is to find a suitable *representation* (in the form of a subset) of the efficient set to present to the DM instead. This problem is not new, and has been considered at least as early as 1980, for example by Steuer and Harris (1980). Since then it has received considerable attention in various areas of multiobjective optimization, including nonlinear optimization (Faulkenberg and Wiecek, 2012; Hancock and Mattson, 2013), linear programming (Shao and Ehrgott, 2016), (mixed) integer (non)linear programming (Sylva and Crema, 2007; Masin and Bukchin, 2008; Eusébio et al., 2014), and discrete optimization in general (Hamacher et al., 2007; Vaz et al., 2015).

In this paper we consider the representation problem within the context of mixed integer programming. Mixed integer programming has become a successful and much accepted methodology for the modelling and solution of challenging combinatorial optimization problems in many different application areas. Its ease of implementation, along with the ever-growing power of modern MIP solver technology (see Achterberg and Wunderling (2013), for example), makes it an effective tool for optimization. These characteristics of mixed integer programming can be exploited in so-called *criterion space search*, which impose only the minimal requirement that a black-box MIP solver be available. Recently a case was made for the effectiveness of using criterion space search over decision space search for the problem of generating the entire efficient set of MIPs with two and three objectives by Boland et al. (2015a,b, 2016a,b).

When it comes to biobjective mixed integer programming, existing methods for generating representations pose one of two drawbacks. First of all, filtering methods that require the entire efficient set as input (Vaz et al. (2015), for example) can be extremely inefficient due to the fact that the efficient set is in many cases exponential in size and generating it is in general an NP-hard problem (Ehrgott et al., 2016). The alternative is generating efficient solutions until a stopping criterion is met, which is usually a certain quality level according to a specified measure of how good a representation is (Hamacher et al., 2007; Sylva and Crema, 2007; Masin and Bukchin, 2008; Eusébio et al., 2014). The drawback in this case is that this quality level (usually a real value) has to be specified a priori by the DM, and it might not be intuitive to a DM what value to choose. It is, on the other hand, a much simpler task for the DM to only have to specify a desired *number* of solutions that they are willing to consider, and then to generate a high quality representation of the specified cardinality. As we shall show in this paper, existing methods do not always produce satisfactory results when specifying a fixed cardinality instead of a desired quality level.

In this paper we present an effective method to generate representations consisting of uniformly spaced nondominated points for biobjective discrete optimization problems given a fixed cardinality of the representation. The method borrows ideas from *Voronoi diagrams* in an attempt to find points that are equidistant in the criterion space. We show on a set of benchmark biobjective MIP instances from the literature that, subject to a fixed cardinality of the representation, our method outperforms existing methods with respect to two standard and accepted quality measures from the literature, namely *coverage* and *uniformity*. Moreover, we prove that a dual relationship exists between the problems of minimizing the coverage error and maximizing the uniformity level, which provides us with an optimality gap. We show that our method is capable of finding solutions that exhibit relatively small optimality gaps, and that in many cases it is able to find the optimal solution.

The rest of the paper is organized as follows. In Section 2 we discuss how to measure the quality of a representation and which quality measures we adopt in this paper, in light of which a formal definition of the problem considered in this paper is given in Section 3. This is followed in Section 4 by a brief survey of existing methods for generating representations. We present a new method in Section 6 and in Section 7 we present a computational study where we illustrate the dominance of the new method in comparison to existing methods. The paper closes with a summary and ideas for future work in Section 8.

2 Quality measures for representations

Apart from simply providing the DM with an arbitrary set of alternatives to choose from, a representation can also, if chosen wisely, provide the DM with an accurate impression of the nature of the tradeoffs that exist among the different objectives. This begs the question of how to measure the *quality*, i.e. the representative power, of a set of efficient solutions. For the biobjective case, visualizing a representation gives an intuitive idea of what a high quality representation might look like. Figure 1 (left) shows the nondominated set of an instance¹ from a class of biobjective mixed integer programming problems introduced by Mavrotas and Diakoulaki (1998) (where the objectives are to be minimized), while Figure 1 (right) shows a representation consisting of five points from this set. Visually an argument can be made for it being a good representation, since the points look roughly equally spaced and they roughly follow the shape of the nondominated frontier. The important question is how to quantify these qualities.

Many different measures have been put forward to measure the quality of representations. Faulkenberg and Wiecek (2010) present a comprehensive survey of more than twenty different

¹This is the first instance in the class “First Problem/C20” of biobjective mixed integer programming instances available at <http://ogma.newcastle.edu.au:8080/vital/access/manager/Repository/uon:13218>.

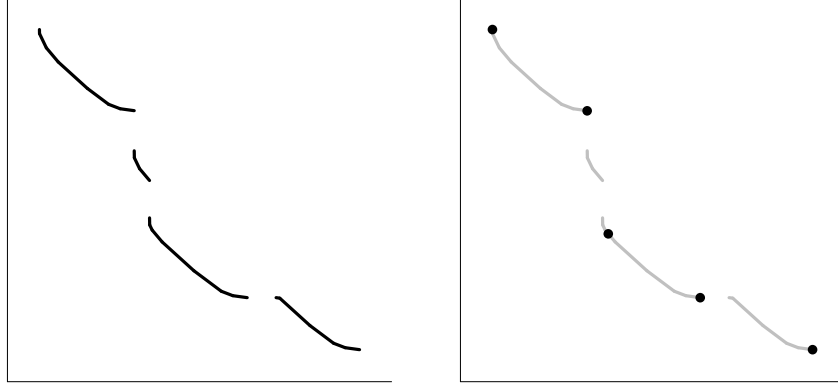


Figure 1: The nondominated frontier (left) and a representation of it (right) for a biobjective MIP, where both objectives are to be minimized. The axes have been scaled in such a way that the left most point is the point $(0, 1)$, while the right most point is the point $(1, 0)$, in order for the objective function values to be of the same order of magnitude.

ones. However, the approach originally proposed by Sayin (2000) is in our opinion the most intuitive, and has become a standard way of evaluating representations (Eusébio et al., 2014; Vaz et al., 2015; Kuhn and Ruzika, 2016; Shao and Ehrgott, 2016). We adopt this approach in this paper as well. Sayin (2000) proposed three criteria of importance when evaluating representations, namely *coverage*, *uniformity* and *cardinality*. Informally speaking, coverage relates to how well regions of nondominated points in the criterion space are represented, and a representation with good coverage would imply that any nondominated point outside the representation is close enough to (i.e. covered by) a point inside the representation. Uniformity relates to the spread of points in the criterion space, and a representation with good uniformity would not contain points that are clustered (and thereby redundant). Finally, cardinality simply relates to the fact that a representation should contain as few points as possible while providing the DM with an adequate impression of the tradeoffs that exist between the different objectives. Note that coverage and uniformity requires a distance metric as a function of pairs of points in the criterion space, which is usually taken to be a p -norm distance metric with either $p = 1, 2$ or ∞ (the *Manhattan*, *Euclidean*, and *Chebyshev* distance metrics, respectively (Sayin, 2000)). The formal definitions of the coverage and uniformity of a representation will be presented in the following section.

3 Formal problem definition

We consider a biobjective mixed integer program (BOMIP), where we minimize over the set of objective function values

$$\mathcal{Y}_{C,A,b,I} = \{y = Cx : Ax \geq b, x \geq 0, x_i \text{ integer } \forall i \in I\} \subset \mathbb{R}^2, \quad (1)$$

corresponding to feasible solutions to the problem, where x is a vector of decision variables of length n , C is a $2 \times n$ matrix, A is an $m \times n$ matrix, b is a vector of length m , and I is the index set of the integer variables. If the values of C, A, b and I are clear from the context, the shorthand \mathcal{Y} will be used instead of $\mathcal{Y}_{C,A,b,I}$. We will assume without loss of generality that $Cx \geq 0$ for all x , and we will refer to \mathbb{R}^2 as the *criterion space*.

Since each element $y = (y_1, y_2) \in \mathcal{Y}$ is a point in \mathbb{R}^2 , minimizing over \mathcal{Y} is defined as finding all nondominated points in \mathcal{Y} , where a point $y \in \mathcal{Y}$ *dominates* another point $y' \in \mathcal{Y} \setminus \{y\}$, if $y_1 \leq y'_1$ and $y_2 \leq y'_2$. A point $y \in \mathcal{Y}$ is *nondominated* if there exists no point $y' \in \mathcal{Y} \setminus \{y\}$ that dominates y , and *weakly nondominated* if there exists no point $y' \in \mathcal{Y} \setminus \{y\}$ such that $y'_1 < y_1$ and $y'_2 < y_2$. The set of all nondominated points in \mathcal{Y} , known as the *nondominated set*, is denoted

by $\mathcal{N}(\mathcal{Y})$, or \mathcal{N} if \mathcal{Y} is clear from the context. For a more complete overview of the formal definitions of these and other concepts in multiobjective optimization, we refer the reader to Ehrgott (2005).

An example of the nondominated set of a BOMIP is given in Figure 2. It has been shown that \mathcal{Y} is the union of a collection of closed and convex polyhedra if the feasible region of x is closed, and so the (possibly nonconvex) nondominated set may contain isolated points, as well as open, half-open and closed line segments (Boland et al., 2015b).

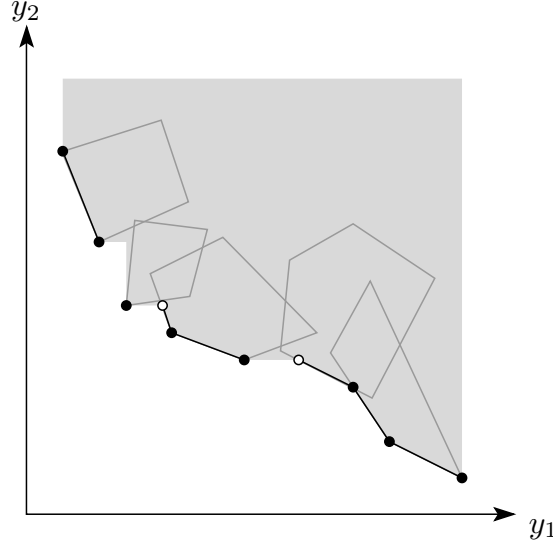


Figure 2: Example of the nondominated set of a BOMIP.

Within the scope of this paper we consider *criterion space search methods* (Boland et al., 2015a), also known as *scalarization methods* (Ehrgott, 2006), and we generate points in the criterion space by solving the problem

$$\mathcal{Y}(\lambda, Y) = \min \{ \lambda y : y \in Y \cap \mathcal{Y} \} \quad (2)$$

where λ is a positive vector of length 2 and $Y \subseteq \mathbb{R}^2$ is a restricted search region in the criterion space. This corresponds to the solution of a single objective minimization problem obtained by scalarizing the two objective functions using weights, while also restricting the values of the objective functions to the region Y in the criterion space. Since $y, \lambda \geq 0$ it is easy to see that the point $\mathcal{Y}(\lambda, Y)$ cannot be dominated by any other point $y \in Y$ within the restricted region.

For mixed integer programs, Problem (2) can be solved as a (single objective) MIP if $Y = \{y \in \mathbb{R}^2 : Dy \geq e\}$, where D is a $l \times 2$ matrix, and e is a vector of length l . Figure 3 shows an example with six points in the nondominated set and the point $\mathcal{Y}(\lambda, Y)$ found using criterion space search, where $\lambda_1 = \lambda_2 = 1$ and $Y = \{y \in \mathcal{Y} : y_2 \leq y_1, y_1 \leq \epsilon\}$. As the example in Figure 3 shows, a point $\mathcal{Y}(\lambda, Y)$ generated during criterion space search is by definition *locally* nondominated within the restricted region Y , but not necessarily *globally* nondominated.

For each pair (λ, Y) a single point $\mathcal{Y}(\lambda, Y) \in \mathcal{Y}$ is thus generated in the criterion space, and many well-known criterion space search methods (e.g. ϵ -constraint methods, the weighted sum method, the balanced box method, etc.) generate points iteratively by varying the values of λ and Y using different strategies, while taking care in making sure that all points generated are (globally) nondominated. Ehrgott (2006) may be consulted for an overview of these and more general criterion space search methods.

An important way in which Problem (2) will be utilized in the methods presented in this paper is to generate a nondominated point within a restricted region in the criterion space that attains a minimal value for at least one of the objectives among all other points in the same

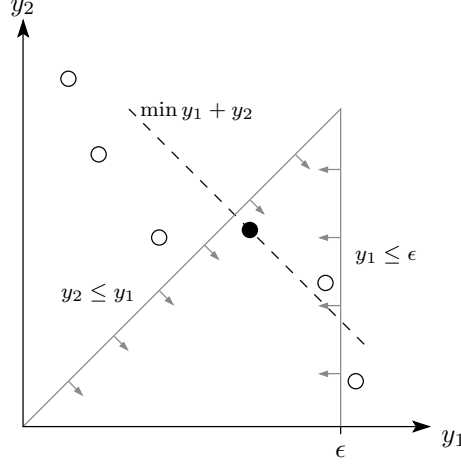


Figure 3: Example of generating a locally nondominated point $\mathcal{Y}(\lambda, Y)$ using criterion space search, where $\lambda_1 = \lambda_2 = 1$ and $Y = \{y \in \mathcal{Y} : y_2 \leq y_1, y_1 \leq \epsilon\}$.

region. The points within the region Y that minimize the first and second objectives respectively are given by

$$\mathcal{Y}^1(Y) = \min \{y_2 : y_1 \leq \mathcal{Y}([1 \ 0], Y), y \in Y \cap \mathcal{Y}\}$$

and

$$\mathcal{Y}^2(Y) = \min \{y_1 : y_2 \leq \mathcal{Y}([0 \ 1], Y), y \in Y \cap \mathcal{Y}\}.$$

In other words each one is obtained by solving a *lexicographical minimization problem*, where, in obtaining $\mathcal{Y}^1(Y)$ for example, the first objective is first minimized without taking the second objective into account at all, followed by minimizing the second objective subject to the constraint that the value of the first objective may not be degraded. In this paper we will refer to $\mathcal{Y}^1(Y)$ as the *left-most point* in the region Y , and $\mathcal{Y}^2(Y)$ as the *right-most point* in the region (i.e. in line with the standard visual representation of the criterion space used in this paper). Figure 4 illustrates a lexicographical minimization problem using criterion space search, where the point $\mathcal{Y}^1(\mathbb{R}^2)$ is generated.

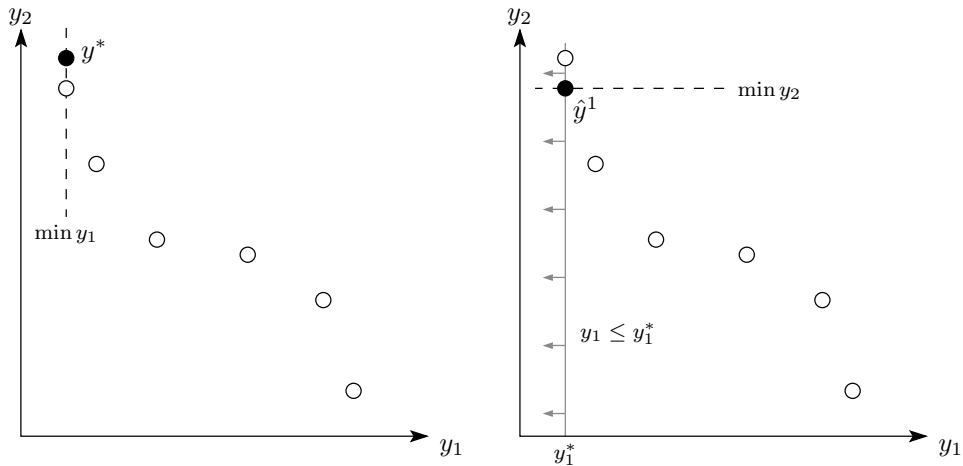


Figure 4: Illustration of how the point $\hat{y}^1 = \mathcal{Y}^1(\mathbb{R}^2)$ may be obtained. In a first step y_1 is minimized, where the resulting point y^* is a weakly nondominated point. In order to ensure that a nondominated point is obtained, a second step is necessary.

Two points in \mathcal{N} that are essential to the methods proposed in this paper are the two

nondominated points that each globally minimizes one of the objectives, i.e. the points $\hat{y}^1 = \mathcal{Y}^1(\mathbb{R}^2)$ and $\hat{y}^2 = \mathcal{Y}^2(\mathbb{R}^2)$. Most criterion space search methods are initialized with these two points (Boland et al., 2015a), as are the methods presented in this paper.

Since coverage and uniformity are defined with respect to the distances between points, the measures for these criteria are functions of a distance metric $d : \mathcal{Y}^2 \mapsto \mathbb{R}$, which usually takes the form of the p -norm

$$d_p(y, y') = \begin{cases} \sqrt[p]{|y_1 - y'_1|^p + |y_2 - y'_2|^p} & \text{for } p = 1, 2, \dots \\ \max(|y_1 - y'_1|, |y_2 - y'_2|) & \text{for } p = \infty, \end{cases}$$

the most common values chosen for p being 1, 2 and ∞ (the Manhattan, Euclidean and Chebyshev distance metrics, respectively). Note that in order to use the p -norm distance metric it is important that the values of both objective functions are of the same order of magnitude, and so we assume that the objective functions have been normalized such that $\hat{y}^1 = [0, 1]$ and $\hat{y}^2 = [1, 0]$. Normalization may be done after obtaining the left-most and right-most points \hat{y}^1 and \hat{y}^2 , and replacing $y = Cx$ in (1) with

$$y_1 = \frac{c_1 x - \hat{y}_1^1}{\hat{y}_1^2 - \hat{y}_1^1} \quad \text{and} \quad y_2 = \frac{c_2 x - \hat{y}_2^2}{\hat{y}_2^1 - \hat{y}_2^2},$$

where c_1 and c_2 are the rows of C .

Coverage is normally measured with respect to points inside the representation as well as points outside it, and so for evaluating coverage the nondominated set \mathcal{N} is required to be available. For a set $S \subseteq \mathbb{R}^2$ and a choice of distance metric d_p , the *coverage error* of the set is defined as

$$\gamma_p(S) = \max_{y \in \mathcal{N}} \min_{y' \in S} d_p(y, y'),$$

i.e. $\gamma_p(S)$ is the largest value such that each point in the nondominated set is within a distance of $\gamma_p(S)$ from a point in the set S . This measure is to be minimized, and so the *coverage problem* is the optimization problem

$$\begin{aligned} & \min \gamma_p(R) \\ & \text{s.t. } R \subset \mathcal{N} \\ & |R| = \psi \end{aligned} \tag{3}$$

for a specified distance metric and cardinality ψ .

The *uniformity level*, on the other hand, does not rely on the nondominated set being available, and is defined as

$$\delta_p(S) = \min_{y, y' \in S, y \neq y'} d_p(y, y'),$$

i.e. the distance between the two closest points in the set. This measure is to be maximized, and so the *uniformity problem* is the optimization problem

$$\begin{aligned} & \max \delta_p(R) \\ & \text{s.t. } R \subset \mathcal{N} \\ & |R| = \psi \end{aligned} \tag{4}$$

for a specified distance metric and cardinality ψ .

Throughout this paper the points in a representation will always be sorted and indexed according to the value of the first objective function, i.e. $R = \{y^1, y^2, \dots, y^{|R|}\} \subset \mathcal{N}$ where $y_1^j < y_1^{j+1}$ for all $j = 1, \dots, |R| - 1$. When measuring uniformity, we are essentially interested in the distances between consecutive points in R , namely the set of distances $\mathcal{D}_p(R) = \{d_p(y^j, y^{j+1}) : j = 1, \dots, |R| - 1\}$.

We conclude this section with three useful results. The first, which will prove useful a number of times throughout this paper, shows that $\delta_p(R) = \min \mathcal{D}_p(R)$.

Lemma 1 *Let $y, y', y'' \in \mathcal{Y}$ be three points that are mutually weakly nondominated and sorted such that $y_1 \leq y'_1 \leq y''_1$. Then $d_p(y, y') \leq d_p(y, y'')$.*

A proof of this result may be found in [Vaz et al. \(2015\)](#). By this lemma it is easy to see that if $R = \{y^1, y^2, \dots, y^{|R|}\}$ is a solution to Problem (3) and if $y^1 = \hat{y}^1$, then a representation with the same or smaller coverage error can be obtained by replacing \hat{y}^1 with any point inbetween it and y^2 . Similarly, if R is a solution to Problem (4) and $y^1 \notin R$, then a representation with the same or larger uniformity level can be obtained by replacing y^1 by \hat{y}^1 . These observations give rise to the following two results.

Lemma 2 *There exists an optimal solution to Problem (3) that contains neither \hat{y}^1 nor \hat{y}^2 .*

Lemma 3 *There exists an optimal solution to Problem (4) that contains both \hat{y}^1 and \hat{y}^2 .*

4 Review of existing methods

Since there is a large amount of literature on generating representations in multiobjective optimization, we review only those methods that are applicable to the specific case we consider, namely criterion space search methods that rely on a black-box MIP solver to generate nondominated points for a BOMIP iteratively without prior knowledge of the nondominated set. We refer to such methods as *insertion methods*, i.e. methods that build the representation point by point, as opposed to *filtering methods*, i.e. methods that start with the entire nondominated set and reduce the set point by point until a suitable representation remains. So we do not review here filtering methods, nor methods for the case where the entire nondominated set is connected (i.e. continuous from \hat{y}^1 to \hat{y}^2). For more on methods not applicable to our case we refer the reader to the extensive survey by [Faulkenberg and Wiecek \(2010\)](#), and to some papers that have appeared since then, e.g. [Faulkenberg and Wiecek \(2012\)](#); [Hancock and Mattson \(2013\)](#); [Shao and Ehrgott \(2016\)](#) for problems with connected nondominated sets and [Vaz et al. \(2015\)](#) for filtering methods for problems with discrete nondominated sets.

As far as we are aware, only four papers propose insertion methods specifically designed for the representation problem in BOMIPs, namely the works by [Hamacher et al. \(2007\)](#), [Sylva and Crema \(2007\)](#), [Masin and Bukchin \(2008\)](#) and [Eusébio et al. \(2014\)](#).

[Hamacher et al. \(2007\)](#) propose two methods based on the so-called ϵ -constraint method. One is a *box-method*, where during each iteration a collection of non-overlapping rectangular regions (*boxes*) are maintained such that no nondominated point falls outside any of these boxes. A new point is generated by selecting the box with the largest area and using an ϵ -constraint on one of the objectives to cut the box in half (vertically) and finding the right-most point in the region defined by the left half of the box. This point is then added to the current representation, and it is used to create two new smaller boxes as subsets of the original box in such a way that regions of the original box that cannot contain any nondominated points are cut away. The second method is initialized by imposing a series of equidistant ϵ -constraints on one of the objectives, after which the right-most point in each region specified by such a constraint is obtained. The authors define the quality measure of a representation to be the area of the largest box, and the algorithm continues iteratively in this way until the area of the largest box is below a specified threshold. The former of these two methods was later extended by [Boland et al. \(2015a\)](#) to the *balanced* box method, where, when a box is cut in half, a point was not only generated in one half of the box, but also in the other half. This was done in order to create tighter boxes in each step for the purpose of developing an efficient method for generating the entire nondominated set instead of a representation. The box-method was also recently extended to the problem of generating representations for problems with three objectives by [Kuhn and Ruzika \(2016\)](#).

[Sylva and Crema \(2007\)](#) and [Masin and Bukchin \(2008\)](#) independently propose essentially identical methods for MIPs with two (or more) objectives, where during each iteration the point

that maximizes the ∞ -norm distance to the current dominated region (defined by the current representation) is generated and added to the representation. The method requires two phases, where in the first phase a point is found that maximizes this distance, while the second phase ensures that a nondominated point is found within the same distance (since the first phase cannot guarantee this). The algorithm continues in this way until the distance of the point that is furthest away is within a specified threshold. The two methods only differ in the way that the distance measure is modelled within the MIP, and even though the methods therefore produce the same representation, they might not have the same performance in terms of computation time required. As far as the authors know a direct comparison has not yet been made.

Finally, Eusébio et al. (2014) consider the representation problem in biobjective integer network flow problems. They propose two methods, one for the coverage problem using the Euclidean distance and one for the uniformity problem using *vertical* distance (where distance is measured as the difference in value of only one objective). In each iteration of the former method the two points in the current representation that are furthest away from each other are selected, and then the rectangle of which they form the corner points is split (horizontally) in half by an ϵ -constraint, and the left-most point in the bottom half of the rectangle is generated. In each iteration of the latter, an ϵ -constraint is imposed a (vertical) distance of some prespecified uniformity below the point in the current representation with lowest (vertical) objective function value. The methods terminate after a desired level of coverage and uniformity, respectively, is attained.

Insertion methods, such as the ones reviewed above and the one to be introduced in the next section, are somewhat limited (greedy approaches as they essentially are) when it comes to generating high quality representations with respect to coverage and uniformity. To see this, note that given a representation $R = \{y^1, y^2, \dots, y^{|R|}\}$ where $|R| \geq 2$, $y^1 = \hat{y}^1$ and $y^{|R|} = \hat{y}^2$, any nondominated point y^* added to this set will fall in the region

$$Y(y^j, y^{j+1}) = \{y \in \mathbb{R} : y_1^j \leq y_1 \leq y_1^{j+1}, y_2^{j+1} \leq y_2 \leq y_2^j\}$$

for some $j \in \{1, 2, \dots, |R| - 1\}$. In other words, y^* will be inserted *inbetween* two points y^j and y^{j+1} , and if done in a way that is supposed to increase coverage or uniformity, it should lie roughly halfway between y^j and y^{j+1} (this is true for all the methods reviewed above). However, this leads to two shorter distances $d_p(y^j, y^*)$ and $d_p(y^*, y^{j+1})$, while the distances $d_p(y^k, y^{k+1})$ for all $k \neq j$ remain roughly twice as large. This results in a set of points that is not uniformly spaced, and it is highly likely that there are much better representations of the same cardinality with respect to coverage and uniformity. In order to maintain a good level of coverage and uniformity, a point should not only be added in the region between one pair of points in R , but in the region between all pairs of points. In other words, in each iteration of an insertion method, $|R| - 1$ points should be inserted to a representation R in order to maintain a balanced set of points. By mathematical induction it is easy to show that $|R| = 2^q + 1$ for some $q \in \mathbb{N}_0$ if built up in this way, given that the insertion method is initialized with $\{\hat{y}^1, \hat{y}^2\}$.

Insertion methods with a quality threshold as a stopping criterion therefore risk terminating the search with large gaps between some points, and the result would be that much better representations of the same cardinality likely exist. On the other hand, insertion methods that take a fixed cardinality as input can overcome this risk by ensuring that the fixed cardinality ψ of the representation is of the form $\psi = 2^q + 1$ for some $q \in \mathbb{N}_0$. The drawback in this case is that the DM is restricted to $\psi \in \{2, 3, 5, 9, 17, \dots\}$ in their choice of the desired cardinality of the representation. However, in Section 7.1 we will discuss a number of ways in which this drawback can be overcome.

5 Bounds for coverage and uniformity

Before presenting a new method in the next section, we first derive some duality results. More specifically, we show that the coverage and uniformity problems can be seen as duals of one another. This is shown by considering what we call *equidistant* representations, a concept which also forms the basis for the method presented in the next section. We call a representation R *equidistant* if any two consecutive points is some fixed distance apart, in other words if $d_p(y^j, y^{j+1}) = d_p(y^k, y^{k+1})$ for all $1 \leq j, k \leq |R| - 1$. Moreover, if $\hat{y}^1 \in R$ and $\hat{y}^2 \in R$ for an equidistant representation R , we call R a *complete* equidistant representation. For an equidistant representation R , we denote by d_R the fixed distance between consecutive points.

The following result provides a useful sufficient condition for optimal uniformity, the proof of which can be found in Appendix 8.

Theorem 1 *If R is a complete equidistant representation for a biobjective optimization problem, then R is an optimal solution to the uniformity problem (4), and the optimal uniformity level is d_R .*

Figure 5 illustrates for a small example why this is true. Given a representation $R \supset \{\hat{y}^1, \hat{y}^2\}$, the points of another representation \tilde{R} can only lie in the regions inbetween the points in R since \tilde{R} only contains nondominated points. If R is equidistant, then at least one pair of points in \tilde{R} will be within a distance of d_R of one another.

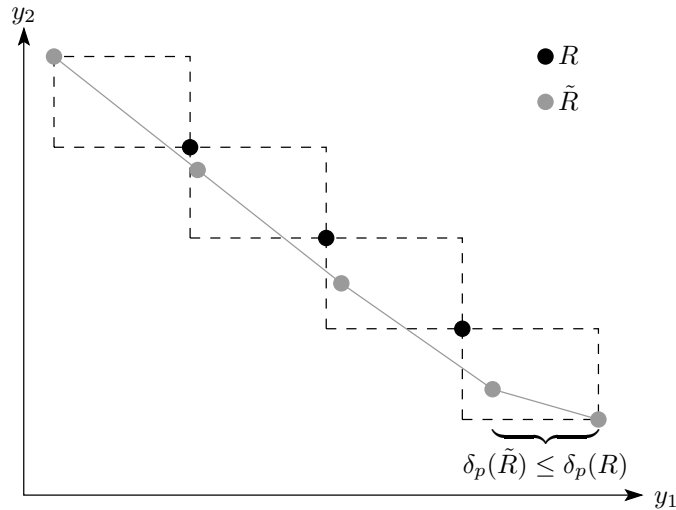


Figure 5: A complete equidistant representation R is optimal in terms of uniformity among all representations of the same cardinality.

It is easy to verify that a complete equidistant representation of a given cardinality always exists if the nondominated set is connected. The implication of the Theorem 1 is therefore that, for problems with connected nondominated sets, finding a representation of cardinality ψ that maximizes uniformity is equivalent to finding an equidistant representation of cardinality ψ .

A similar result can be proved for coverage as follows.

Theorem 2 *If $R' = \{y'^1, \dots, y'^{|R'|}\}$ is a complete equidistant representation of odd cardinality, then*

$$R = \bigcup_{j=1, \dots, \frac{|R'|-1}{2}} \{y'^{2j}\}$$

is an optimal optimal solution to the coverage problem (3), and the optimal coverage error is $d_{R'}$.

Figure 6 illustrates for a small example why this is true. Given a representation R , the points of another representation \tilde{R} can only lie in the regions inbetween the points in $R \cap \{\hat{y}^1, \hat{y}^2\}$ since \tilde{R} only contains nondominated points. If R is constructed as in Theorem 2, then at least one nondominated point outside the set \tilde{R} will lie beyond a distance of $\gamma_p(R)$ from the closest point in the set \tilde{R} .

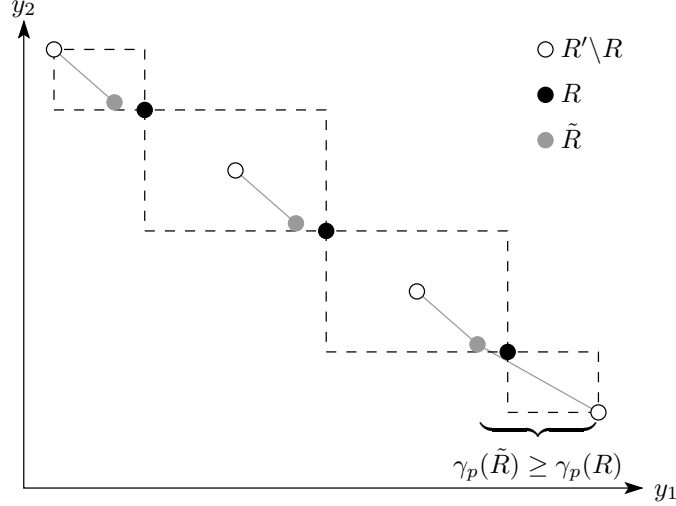


Figure 6: A complete equidistant representation R' contains a representation $R \subset R'$ that is optimal in terms of uniformity among all representations of the same cardinality.

The implication of this theorem is that, for problems with connected nondominated sets, finding a representation of cardinality ψ that minimizes the coverage error is equivalent to finding an equidistant representation of cardinality $2\psi + 1$. We thus have the following duality result.

Corollary 1 *If \mathcal{N} is connected and $R \subset \mathcal{N}$ is a complete equidistant representation with $|R| = 2\psi + 1$, then*

$$\max_{R' \in \mathcal{N}, |R'|=2\psi+1} \delta_p(R') = d_R = \min_{R' \in \mathcal{N}, |R'|=\psi} \gamma_p(R').$$

For problems with disconnected nondominated sets a similar result can be derived. Let \mathcal{N} be disconnected and let $\mathcal{W} \supseteq \mathcal{N}$ be the set of all weakly nondominated points, and note that \mathcal{W} does not contain any open line segments. The *connected relaxation* of \mathcal{N} is, as illustrated by Figure 7, the connected set

$$\tilde{\mathcal{N}} = \mathcal{W} \bigcap_{(y', y'') \in \mathcal{B}} [y', y'']$$

where $[y, y']$ denotes the set of all points on the straight line segment between the points $y, y' \in \mathbb{R}^2$ and where

$$\mathcal{B} = \{(y, y') \in \mathcal{W} \times \mathcal{W} : Y(y, y') \cap \mathcal{W} = \{y, y'\}\}.$$

Since $\mathcal{N} \subseteq \tilde{\mathcal{N}}$, solving the coverage and uniformity problems on $\tilde{\mathcal{N}}$ provides dual bounds for the same problems on \mathcal{N} . Since $\tilde{\mathcal{N}}$ is connected, we therefore have the following result by Corollary 1.

Corollary 2 *For a biobjective optimization problem with a disconnected nondominated set \mathcal{N}*

$$\max_{R \subset \mathcal{N}, |R|=2\psi+1} \delta_p(R) \leq \min_{R \subset \mathcal{N}, |R|=\psi} \gamma_p(R).$$

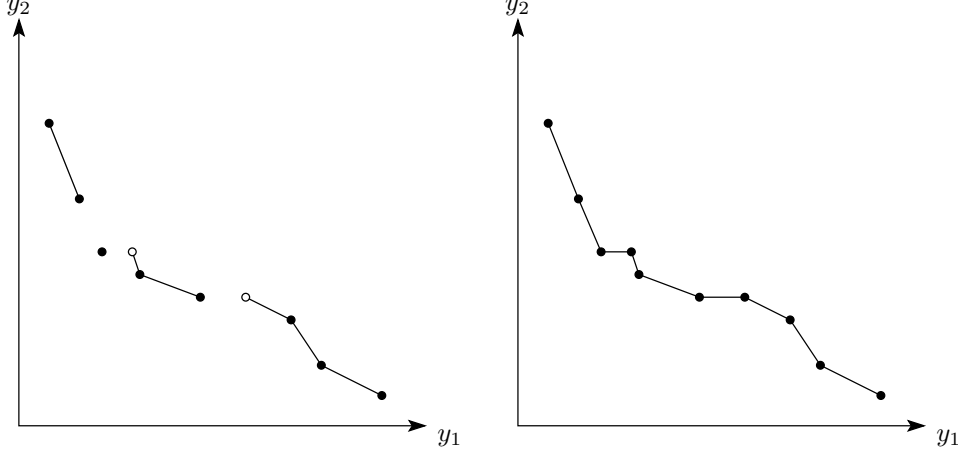


Figure 7: Constructing the connected relaxation of the nondominated set of a BOMIP.

6 A new method

In this section we present the *Voronoi cut method*, which is based on the concept of a *Voronoi diagram* from computational geometry (de Berg et al., 2000). Applications of Voronoi diagrams in operations research are often found in location problems (Okabe and Suzuki, 1997), recent examples of which include location districting (Novaes et al., 2009) and problems related to location density (Cachon, 2014). There are indeed strong similarities between the representation problem in multiobjective optimization and location problems, as noted by Vaz et al. (2015). In fact, the coverage problem corresponds to the so-called k -center problem, while the uniformity problem corresponds to the k -dispersion problem (see Vaz et al. (2015) for further details).

Given two nondominated points y' and y'' such that $y'_1 \leq y''_1$, consider the problem of finding a nondominated point y in the region

$$Y(y', y'') = \{y \in \mathbb{R} : y'_1 \leq y_1 \leq y''_1, y'_2 \leq y_2 \leq y''_2\}$$

such that the representation $R = \{y\}$ is optimal in terms of coverage for this region. By Lemma 1, the point that is furthest away from y will be either y' and y'' regardless of where y lies in this region. The point y is therefore the point that minimizes $\max(d_p(y, y'), d_p(y, y''))$, which is equivalent to stating that it is the point that maximizes $\min(d_p(y, y'), d_p(y, y''))$. The representation $R' = \{y', y, y''\}$ is therefore also optimal in terms of uniformity for this region.

It therefore follows that we have the “local” duality result

$$\max_{R \subset Y(y', y''), |R|=3} \delta_p(R) = \min_{R \subset Y(y', y''), |R|=1} \gamma_p(R) = \min_{y \in Y(y', y'')} \max(d_p(y, y'), d_p(y, y'')).$$

Using this result, we propose employing an iterative greedy method that adds a point to an existing representation R by selecting two consecutive points $y', y'' \in R$ and solving this local problem. The question remains how to find the point y that minimizes $\max(d_p(y, y'), d_p(y, y''))$.

Towards this end we utilize the main idea behind the construction of a Voronoi diagram, namely to divide the search region $Y(y', y'')$ into two distinct cells of points that share the same nearest neighbour among the two points y' and y'' , i.e.

$$V_p^{\leq}(y', y'') = \{y \in Y(y', y'') : d_p(y, y') \leq d_p(y, y'')\}$$

and

$$V_p^{\geq}(y', y'') = \{y \in Y(y', y'') : d_p(y, y') \geq d_p(y, y'')\}.$$

The nondominated point y that maximizes $\min(d_p(y, y'), d_p(y, y''))$ is therefore either the nondominated point in $V_p^{\leq}(y', y'')$ that is furthest away from y' , or the nondominated point in

$V_p^\geq(y', y'')$ that is furthest away from y'' . Once these two points have been obtained, it can be determined by inspection which one maximizes $\min(d_p(y, y'), d_p(y, y''))$.

Figure 8 illustrates this concept for the Euclidean distance metric, for which it is known (and not difficult to verify) that the Voronoi cells are divided by a straight line that is the perpendicular bisector of the line segment between y' and y'' (de Berg et al., 2000). Note that if the nondominated set is continuous across the two Voronoi cells, i.e. if the nondominated set is continuous from a point in $V_p^\leq(y', y'')$ to a point in $V_p^\geq(y', y'')$, then there exists a single nondominated point in the intersection $V_p^\leq(y', y'') \cap V_p^\geq(y', y'')$, i.e. an equidistant point y such that $d_p(y, y') = d_p(y, y'')$.

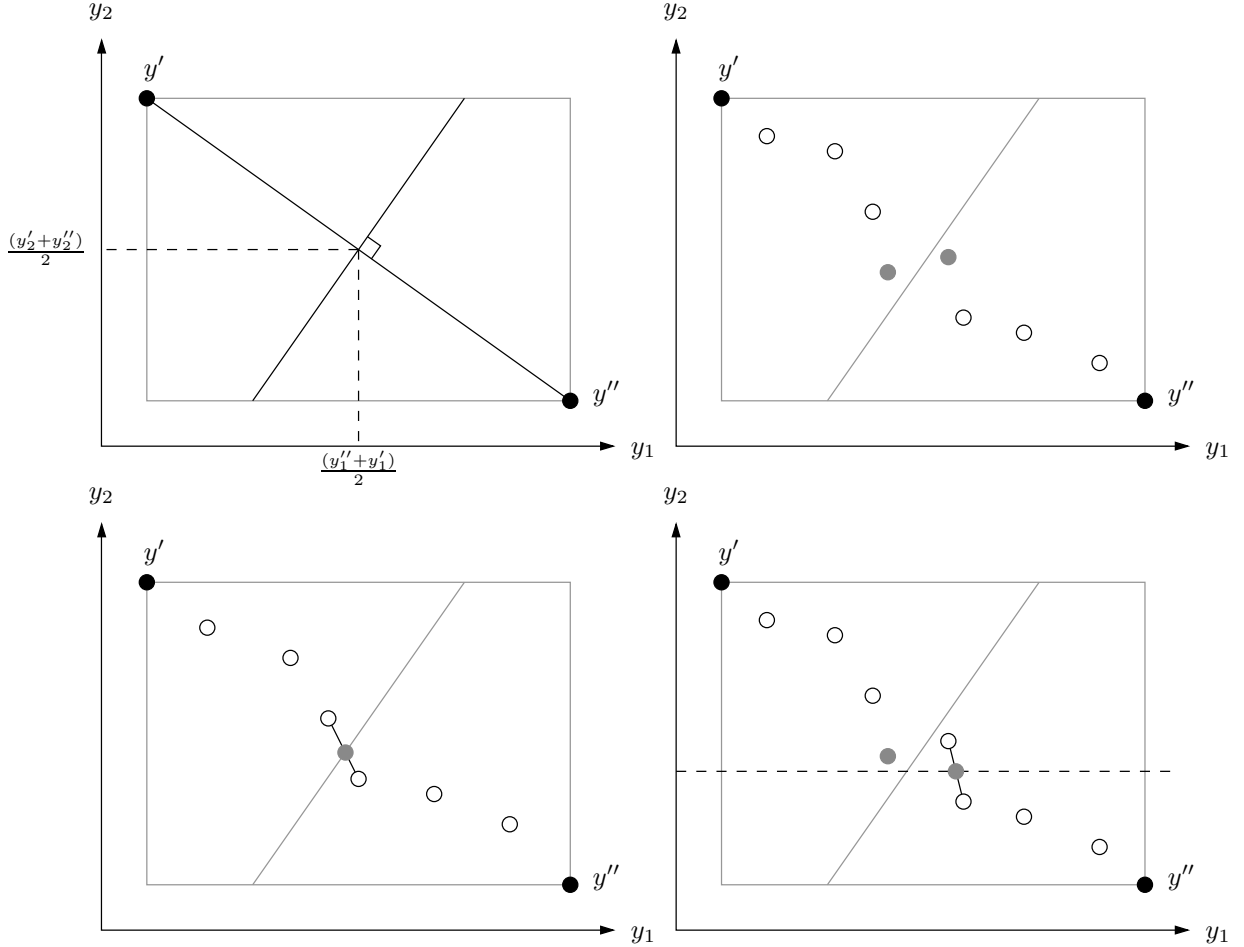


Figure 8: Constructing the Voronoi cells for two points y' and y'' for $p = 2$. The nondominated points between y' and y'' are shown, and in case the cut does not intersect with a continuous part of the nondominated set, one of two points maximizes $\min(d_p(y, y'), d_p(y, y''))$ (top-right). If the cut intersect a continuous part of the nondominated set, a single point (shaded) maximizes $\min(d_p(y, y'), d_p(y, y''))$ (bottom-left). Moreover, a second step is necessary, since the first step does not guarantee a nondominated point (bottom-right).

In order to find the nondominated point in $V_p^\leq(y', y'')$ that is furthest away from y' , it is natural to want to consider (according to Lemma 1) the right-most point in the region $V_p^\leq(y', y'')$, i.e. $\mathcal{Y}^2(V_p^\leq(y', y''))$. However, the right-most point in a restricted region is only guaranteed to be nondominated *within* the region, and not necessarily globally nondominated. The same holds for the left-most point in $V_p^\geq(y', y'')$. Figure 8 (top-right) illustrates how it can happen that, for example, the left-most point in $V_p^\geq(y', y'')$ can be dominated by a point outside this region. The following result, however, guarantees that at least one of the two points is nondominated.

Lemma 4 *Let y' and y'' , $y'_1 < y''_1$, be two nondominated points and let $y^L = \mathcal{Y}^2(V_p^{\leq}(y', y''))$ and $y^R = \mathcal{Y}^1(V_p^{\geq}(y', y''))$. Then (a) not both y^L and y^R are dominated, and (b) if one of them is dominated, it is dominated by the other.*

If, for instance, y^L dominates y^R , then, since only points in the region $V_p^{\leq}(y', y'')$ can dominate y^R and since y^L is the right-most point in $V_p^{\leq}(y', y'')$, there is no point y such that $y_2 < y_2^L$ that dominates y^R . Similarly it can be shown that there is no point y such that $y_1 < y_1^R$ that dominates y^L . The next result therefore provides a means to remedy the situation where one of these points is dominated, as illustrated by Figure 8 (bottom-right).

Corollary 3 *Let y' and y'' , $y'_1 < y''_1$, be two nondominated points and let $y^L = \mathcal{Y}^2(V_p^{\leq}(y', y''))$ and $y^R = \mathcal{Y}^1(V_p^{\geq}(y', y''))$. If y^L dominates y^R , then*

$$\mathcal{Y}^1(V_p^{\geq}(y', y'')) \cap \{y : y_2 \leq y_2^L - \epsilon\}$$

is nondominated for any $\epsilon > 0$. If y^R dominates y^L , then

$$\mathcal{Y}^2(V_p^{\leq}(y', y'')) \cap \{y : y_1 \leq y_1^R - \epsilon\}$$

is nondominated for any $\epsilon > 0$.

During each iteration of the Voronoi cut method the largest gap (according to the distance measure) between any two consecutive points in the current representation is determined, and a nondominated point is inserted in accordance with the observations above. The algorithm stops once a desired number ψ of points have been generated.

In order to apply criterion space search, the regions $V_p^{\leq}(y', y'')$ and $V_p^{\geq}(y', y'')$ need be defined using linear constraints, and in order to facilitate this we focus on the intersection between these two sets, namely

$$V_p^{\leq}(y', y'') \cap V_p^{\geq}(y', y'') = \{y \in Y(y', y'') : d_p(y, y') = d_p(y, y'')\},$$

which we will henceforth refer to the *Voronoi cut* between points y' and y'' . In the following we treat each of the three cases $p = 1, 2, \infty$ separately.

6.1 Voronoi cut for $p = 1$

For $p = 1$ the Voronoi cut is given by points $y \in Y(y', y'')$ satisfying

$$|y_1 - y'_1| + |y_2 - y'_2| = |y_1 - y''_1| + |y_2 - y''_2|$$

which may be simplified to the more convenient form

$$y_1 - \frac{y'_1 + y''_1}{2} = y_2 - \frac{y'_2 + y''_2}{2}$$

using the fact that $y \in Y(y', y'')$. In this case the Voronoi cut is a straight line with a 45° slope bisecting the line segment between y' and y'' . The Voronoi cells may be specified using linear constraints as

$$V_1^{\leq}(y', y'') = \left\{ y \in Y(y', y'') : y_1 - \frac{y'_1 + y''_1}{2} \leq y_2 - \frac{y'_2 + y''_2}{2} \right\}$$

and

$$V_1^{\geq}(y', y'') = \left\{ y \in Y(y', y'') : y_1 - \frac{y'_1 + y''_1}{2} \geq y_2 - \frac{y'_2 + y''_2}{2} \right\}.$$

6.2 Voronoi cut for $p = 2$

We have already noted that for $p = 2$ the Voronoi cut is a straight line, given by

$$\sqrt{(y_1 - y'_1)^2 + (y_2 - y'_2)^2} = \sqrt{(y_1 - y''_1)^2 + (y_2 - y''_2)^2}$$

which may be simplified to the more convenient form

$$\left(y_1 - \frac{y'_1 + y''_1}{2}\right)(y''_1 - y'_1) = \left(y_2 - \frac{y'_2 + y''_2}{2}\right)(y'_2 - y''_2).$$

The Voronoi cells may be specified using linear constraints as

$$V_2^{\leq}(y', y'') = \left\{y \in Y(y', y'') : \left(y_1 - \frac{y'_1 + y''_1}{2}\right)(y''_1 - y'_1) \leq \left(y_2 - \frac{y'_2 + y''_2}{2}\right)(y'_2 - y''_2)\right\}$$

and

$$V_2^{\geq}(y', y'') = \left\{y \in Y(y', y'') : \left(y_1 - \frac{y'_1 + y''_1}{2}\right)(y''_1 - y'_1) \geq \left(y_2 - \frac{y'_2 + y''_2}{2}\right)(y'_2 - y''_2)\right\}.$$

6.3 Voronoi cut for $p = \infty$

For $p = \infty$ we present in this section a procedure to find the right most point in $V_{\infty}^{\leq}(y', y'')$ and the left-most point in $V_{\infty}^{\geq}(y', y'')$ for the case where $y''_1 - y'_1 \geq y'_2 - y''_2$. Visually this corresponds to the case where $Y(y', y'')$ is a rectangle that is wider than it is high. If $y''_1 - y'_1 \leq y'_2 - y''_2$, the procedure is exactly the same, except with the roles of the objective functions (or from a visual point of view, the axes) reversed.

For $p = \infty$, the Voronoi cut is given by

$$\max(y_1 - y'_1, y'_2 - y_2) = \max(y''_1 - y_1, y_2 - y''_2)$$

which may, by considering all four possible outcomes of the two “max” operators, be simplified to a more convenient piecewise linear form. To simplify notation, let

$$s_{\min} = y'_1 + y'_2$$

and

$$s_{\max} = y''_1 + y''_2.$$

The Voronoi cut may be simplified to

$$\begin{aligned} y_1 - y_2 &= y''_1 - y'_2 && \text{if } y_1 + y_2 \leq s_{\min}, \\ y_1 &= \frac{1}{2}(y''_1 + y'_1) && \text{if } s_{\min} \leq y_1 + y_2 \leq s_{\max} \\ y_1 - y_2 &= y'_1 - y''_2 && \text{if } s_{\max} \leq y_1 + y_2. \end{aligned}$$

We can distinguish between three different cases:

- (i) $s_{\max} - s_{\min} = 0$, in which case the Voronoi cut is a straight line given by $y_1 - y_2 = y''_1 - y'_2 = y'_1 - y''_2$, i.e. a line with a 45° slope bisecting the line segment between y' and y'' (see Figure 9, top). In fact, in this case the Voronoi cut is the same for all three cases $p = 1, 2, \infty$.
- (ii) $s_{\max} - s_{\min} \leq \frac{y'_1 + y''_1}{2}$, in which case the Voronoi cut is piecewise linear, as seen in Figure 9 (bottom left). Moreover, the regions $V_{\infty}^{\leq}(y', y'')$ and $V_{\infty}^{\geq}(y', y'')$ are nonconvex and cannot be described by a set of linear constraints $Dy \geq e$.

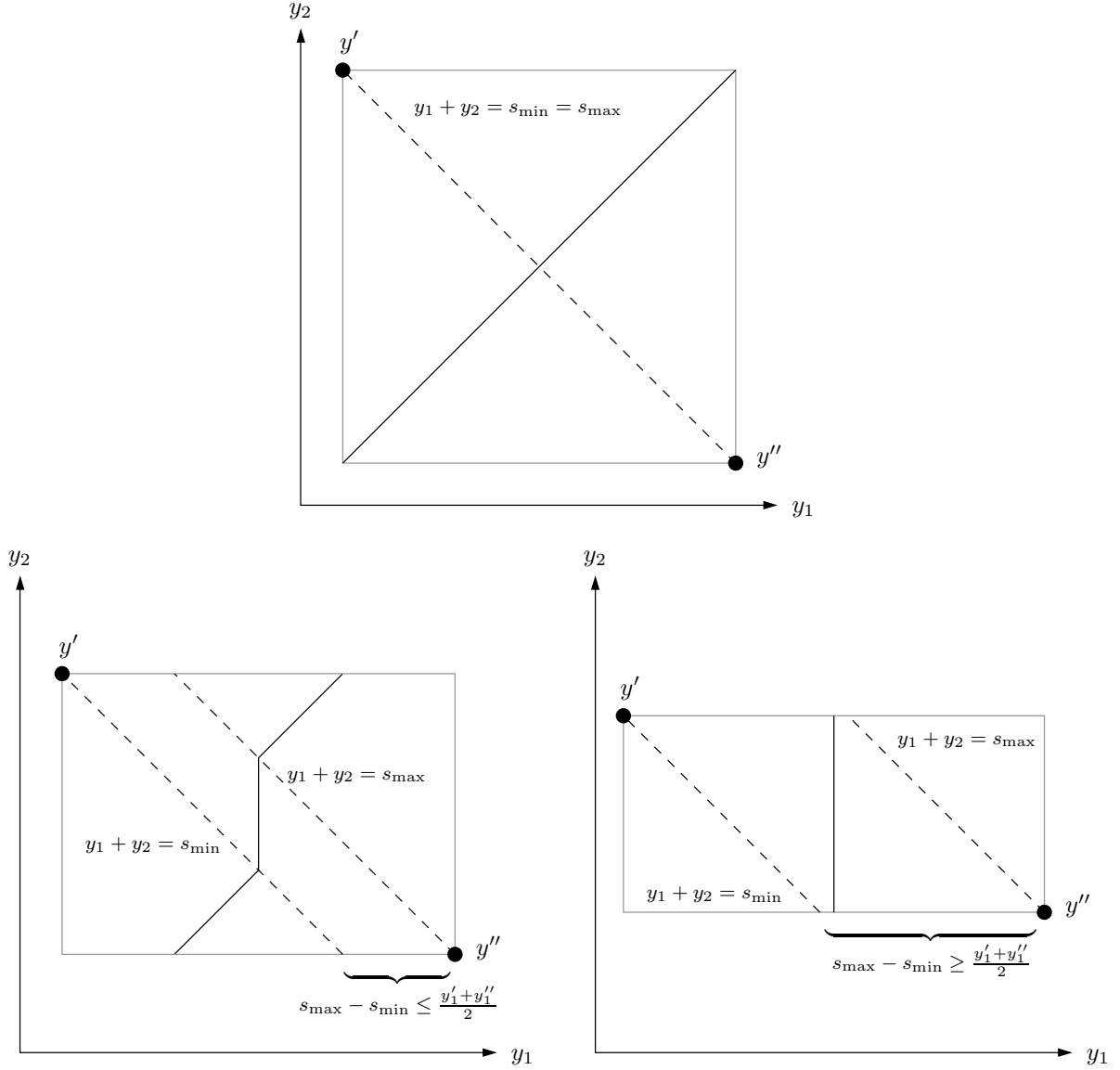


Figure 9: The Voronoi cut for $p = \infty$.

- (iii) $s_{\max} - s_{\min} \geq \frac{y'_1 + y''_1}{2}$, in which case the Voronoi cut is a vertical line exactly halfway between y'_1 and y''_1 , as shown in Figure 9 (bottom right).

Thus, for Case (i), $V_{\infty}^{\leq}(y', y'') = V_1^{\leq}(y', y'')$ and $V_{\infty}^{\geq}(y', y'') = V_1^{\geq}(y', y'')$, while for Case (iii)

$$V_{\infty}^{\leq}(y', y'') = \left\{ y \in Y(y', y'') : y_1 \leq \frac{y'_1 + y''_1}{2} \right\},$$

and

$$V_{\infty}^{\geq}(y', y'') = \left\{ y \in Y(y', y'') : y_1 \geq \frac{y'_1 + y''_1}{2} \right\}.$$

For Case (ii), it is a little bit more complicated since the regions are nonconvex. However, note that in this case $V_{\infty}^{\leq}(y', y'')$ is the union of the convex regions

$$Y_1 = \left\{ y \in Y(y', y'') : y_1 - y_2 \leq y''_1 - y'_2, y_1 \leq \frac{y'_1 + y''_1}{2} \right\}$$

and

$$Y_2 = \left\{ y \in Y(y', y'') : y_1 - y_2 \leq y'_1 - y''_2, y_1 \geq \frac{y'_1 + y''_1}{2} \right\},$$

as shown in Figure 10. In order to find the right-most point in $V_\infty^\leq(y', y'')$, we propose to first find $\tilde{y} = \mathcal{Y}^2(Y_1)$. If $\tilde{y} \leq y''_2 + \frac{y'_1 - y''_1}{2}$, then it dominates all points in Y_2 (see Figure 10, left), and it therefore is the right-most point in $V_\infty^\leq(y', y'')$. Otherwise, there might be a point in the region $Y_2 \cap \{y \in Y(y', y'') : y_2 < \tilde{y}_2\}$ (see Figure 10, right). However, if this region is not guaranteed to contain any feasible points, and as has been noted by Boland et al. (2015a) as well, MIP solvers tend to have a hard time proving infeasibility. In order to ensure that there is at least one feasible point, the region

$$\{y \in Y(y', y'') : y_2 < \tilde{y}_2, y_1 \geq \tilde{y}_1, y_1 - y_2 \leq y'_1 - y''_2\}$$

is considered instead, and the right-most point in this region will then be the right-most point in $V_\infty^\leq(y', y'')$. The same ideas can then also be applied to find the left-most point in $V_\infty^\geq(y', y'')$.

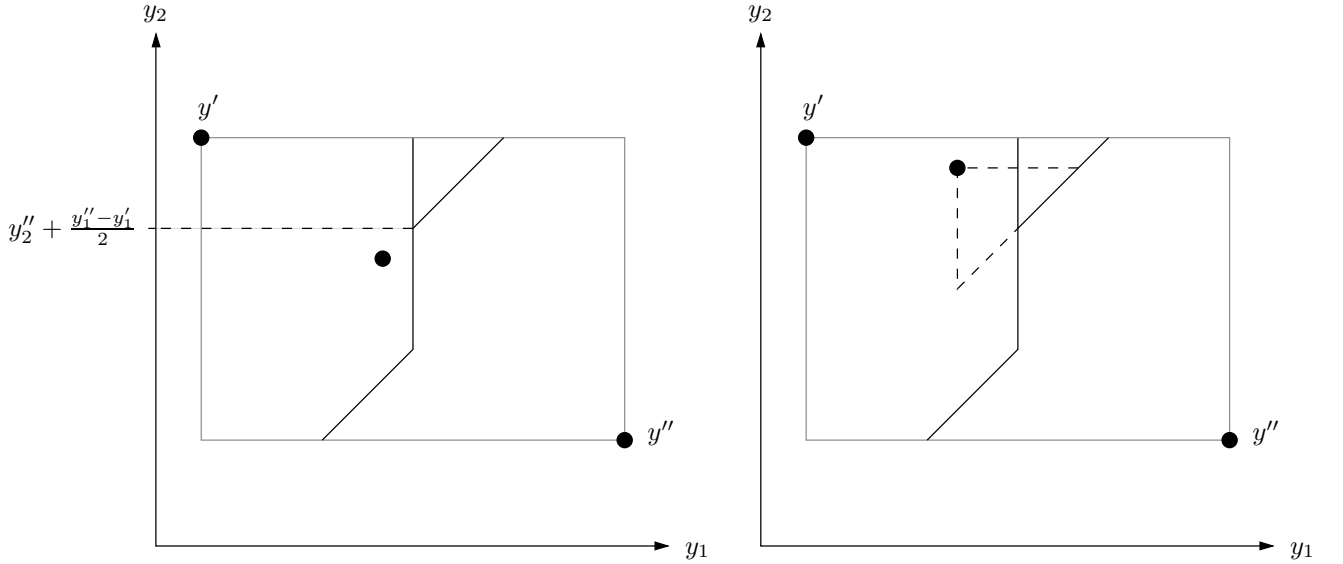


Figure 10: Two phase method for dealing with the nonconvexity of the Voronoi cut for $p = \infty$.

6.4 Computational complexity

The computational complexity of these methods depends on the algorithms used by the solver and the specific optimization problem under consideration, but an impression can be given by considering the worst case number of different calls to the solver required. The Voronoi cut method starts by determining \hat{y}^1 and \hat{y}^2 which requires two lexicographical optimization problems, and thus 4 calls to the solver. For $p = 1, 2$ and for each Voronoi cut, once again two lexicographical optimization problems are solved on either side of the cut, and in the worst case one point can be dominated by another which requires a third lexicographical optimization problem to be solved. For each point that is to be generated, apart from \hat{y}^1 and \hat{y}^2 , 6 calls to the solver are therefore required in the worst case. For generating a representation consisting of ψ points, $4 + 6(\psi - 2)$ calls to the solver are required in the worst case for $p = 1, 2$. For $p = \infty$, finding a point on either side of the cut requires 4 calls in the worst case, plus an additional 2 if one point dominates another, and so $4 + 10(\psi - 2)$ calls to the solver is required in the worst case.

7 Computational results

In this section we compare the performance of the Voronoi cut method with existing methods from the literature, specifically the methods of Hamacher et al. (2007), Sylva and Crema (2007), Masin and Bukchin (2008) and Eusébio et al. (2014). Hamacher et al. (2007) proposed two algorithms, namely an *a posteriori algorithm* and an *a priori algorithm*, which we henceforth abbreviate as *HaPeRu2007(1)* and *HaPeRu2007(2)*, respectively. Similarly, Eusébio et al. (2014) proposed a γ -*representation algorithm* and a δ -*uniform representation algorithm*, which we henceforth abbreviate as *EuFiEh2014(1)*, and *EuFiEh2014(2)*, respectively. Finally, the methods of Sylva and Crema (2007) and Masin and Bukchin (2008) are abbreviated as *SyCr2007* and *MaBu2008*, respectively. We also include the *weighted sum method* (Aneja and Nair, 1979) due to its simplicity and the fact that it is, along with the ϵ -constraint method, one of the most popular criterion search space methods typically utilized.

All of these methods necessarily generate representations that include the points \hat{y}^1 and \hat{y}^2 , and so by Lemmas 2 and 3 they are appropriate for the uniformity problem, but not for the coverage problem. However, Theorem 2 provides a means of constructing from a representation $R \supset \{\hat{y}^1, \hat{y}^2\}$ a representation R' of cardinality $\frac{|R|-1}{2}$ that is appropriate for the coverage problem. In what follows, we define the *sub-coverage error* of R as the coverage error of R' .

In what follows we first discuss the choice of ψ . Thereafter we present computational results for generating five points using the different methods, and we compare performance of the methods with respect to coverage and uniformity for all three distance measures considered.

7.1 Choice of cardinality of the representation

As mentioned in Section 4, when using an insertion method, a fixed cardinality of the form $\psi = 2^q + 1$ for some $q \in \mathbb{N}_0$ is recommended if a representation is desired that is close to being equidistant, since otherwise there might remain large gaps between some points in the final output. However, this leaves cases where the DM desires a number of points that is *not* of this form unresolved. We suggest here a number of ways to overcome this drawback.

Consider the case where the decision maker desires a number of points $\psi = 2^q + 1 + r$ for some $q, r \in \mathbb{N}$. One possible way of generating such a set is to initialize the search with $\{\hat{y}^1, \hat{y}^2\}$ and first generate r points, where the j -th point for all $j = 1, \dots, r$ is a distance of $j \cdot d$ from \hat{y}^1 , given some fixed distance $d \in \mathbb{R}_+$. If y^* is the final point in this sequence, then an insertion method can be used to generate the remaining $2^q - 1$ points between y^* and \hat{y}^2 . This, however, poses two difficulties. First of all, it is not clear what the fixed distance d between the first r points should be. A rough guess could be $d_p(\hat{y}^1, \hat{y}^2)/\psi$, but this will only be close to the true distance between a uniform set of ψ points if the nondominated set roughly follows a straight line in the criterion space, which is usually not the case. Secondly, and most importantly, it is not clear how to generate a point that is some fixed distance d away from a reference point given some distance measure. Note that the Voronoi cut method never attempts to find points within a specific distance, but instead attempts to find points that are *equidistant* from two reference points, regardless of what this distance will be. One approach would be to consider *weighted* Voronoi diagrams, where the Voronoi cut would be all points that are a distance away from one reference point that is a constant multiple of the distance to the other reference point. However, in this case the Voronoi cut is no longer a straight line.

Our recommendation is to instead use the Voronoi cut method in an interactive approach. The Voronoi cut method can be used to generate, for example, five points, after which these solutions are presented to the DM for consideration. If the DM can then identify a region of interest between two specific points, the Voronoi cut method can be used to generate an additional three points inbetween these two. In this case the interactive search can iteratively reduce the region in the criterion space where the DM would prefer to find a solution, while ensuring a managably sized and uniformly spaced set of points is presented to the DM each time.

This process can be repeated until the DM has considered a satisfactory number of points.

7.2 Comparison of methods for $\psi = 5$ and $\psi = 9$

We use four classes of BOMIP instances that were also used in Boland et al. (2015b) and which are available online at <http://ogma.newcastle.edu.au:8080/vital/access/manager/Repository/uon:13218>. These consist of two classes of integer programming problems, namely 20 instances of biobjective assignment problem (AP) and 20 instances of the two-dimensional knapsack problem (2DKP), and two classes of mixed integer programming problems, namely 12 instances of the biobjective uncapacitated facility location problem (BUFLP) and 25 instances of a class of mixed integer programming problems introduced by Mavrotas and Diakoulaki (1998) (MILP). Since we consider three different distance measures and two cardinalities, this amounts to 462 instances in total. All methods were implemented in C++ using CPLEX 12.6 as the black-box MIP solver on a 2.20GHz Intel(R) Core(TM) i3-2330M CPU with 4 GB RAM. The default settings for CPLEX were used, except for the relative MIP gap tolerance, which was set to 1×10^7 .

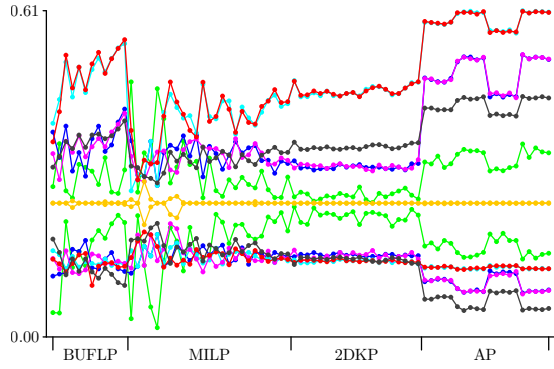
Figure 11 shows, for different values of ψ and p , the effectiveness of the Voronoi cut method compared to other methods from the literature. Two points are plotted for each method and each instance, one for the uniformity level and one for the sub-coverage error of the representation generated by the method (note that by Corollary 2 the former point is always below the latter). As can be seen, in most cases the Voronoi cut method outperforms all other methods with respect to both the coverage and uniformity problems, and it attains relatively small optimality gaps. In particular, the Voronoi cut method was able to close the gap in 57 out of 462 cases, whereas among all the other methods only *MaBu2008/SyCr2007* was able to close the gap, and only for one instance (instance 1 from the class BUFLP).

In terms of computation times, the average time per call to the solver varies little across the different methods, as can be seen from Table 1. An exception to this is the methods *Sylva and Crema (2007)* and *Masin and Bukchin (2008)* when applied to instances of the class AP, where the objective function used by this method slows the solver down considerably.

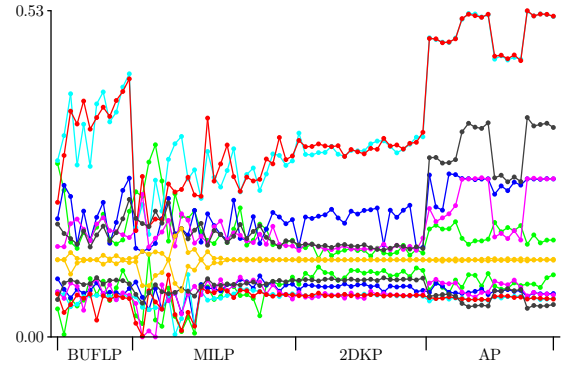
		Avg time per CPLEX call			
		BUFLP	MILP	2DKP	AP
$\psi = 5$	Weighted sum	0.06	0.05	0.94	1.28
	HaPeRu2007(1)	0.20	0.27	1.25	2.43
	HaPeRu2007(2)	0.19	0.24	1.18	2.18
	SyCr2007	0.23	0.26	1.96	59.20
	MaBu2008	0.27	0.25	1.96	66.59
	EuFiEh2014(1)	0.19	0.20	1.20	2.43
	EuFiEh2014(2)	0.15	0.19	1.19	2.17
	Voronoi Cut	0.40	0.19	1.71	4.02
$\psi = 9$	Weighted sum	0.05	0.04	0.88	1.19
	HaPeRu2007(1)	0.22	0.27	1.16	2.75
	HaPeRu2007(2)	0.22	0.25	1.12	2.29
	SyCr2007	0.31	0.34	2.51	124.93
	MaBu2008	0.34	0.32	2.70	88.93
	EuFiEh2014(1)	0.23	0.24	1.12	2.79
	EuFiEh2014(2)	0.19	0.21	1.13	2.24
	Voronoi Cut	2.25	0.23	1.67	4.28

Table 1: For each method and each class of instances, the average computation time in seconds per single call to CPLEX over all instance in the class.

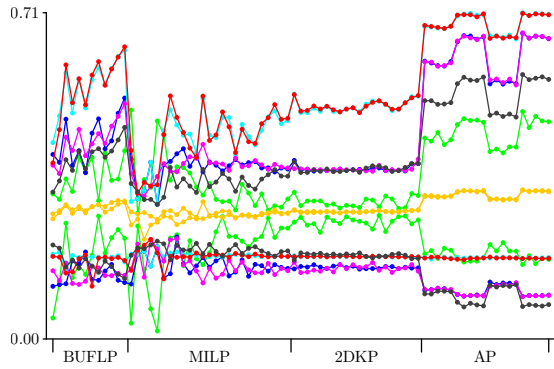
What is more of interest is the number of times the solver is called by each method. All



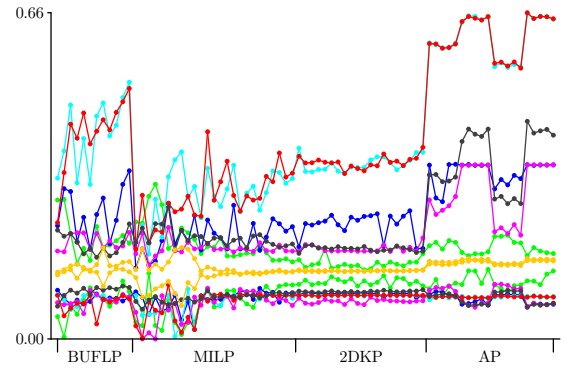
(a) $p = 1, \psi = 5$



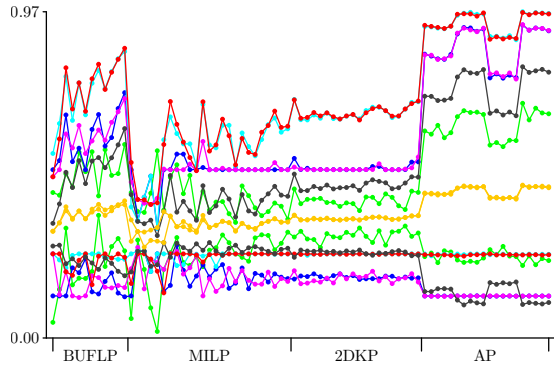
(b) $p = 1, \psi = 9$



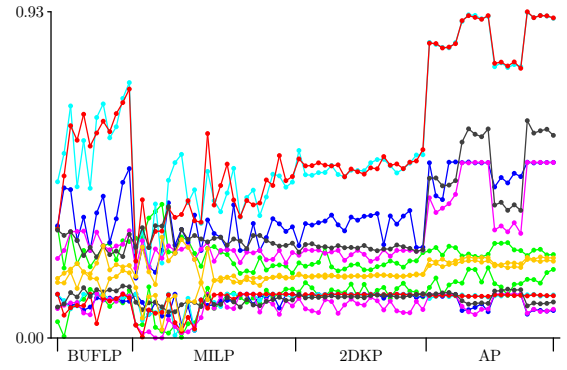
(c) $p = 2, \psi = 5$



(d) $p = 2, \psi = 9$



(e) $p = \infty, \psi = 5$



(f) $p = \infty, \psi = 9$

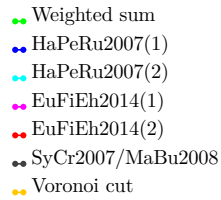


Figure 11: For different methods and values of ψ and p , two points are plotted for each instance showing the uniformity level and sub-coverage error, respectively.

reviewed methods, except for the weighted sum method, require two calls for each point to be generated, i.e. 2ψ calls in total. The weighted sum method requires two calls for the top-left and bottom-right points, and then only one call for each remaining point, i.e. $4 + \psi - 2$ calls in total. For the Voronoi cut method, the number of calls may vary. Table 2 shows the theoretical worst case number of calls to the solver required by the Voronoi cut method, and for each class of instances the maximum number of calls that was actually performed. As can be seen, the high accuracy of the Voronoi cut method seen in the previous section comes at the cost of requiring more calls to the solver — slightly more than twice as many calls for the considered instances for $\psi = 5, 9$.

	Worst case # of calls for Voronoi cut method			Max # of calls							
				$\psi = 5$				$\psi = 9$			
	$\psi = 5$	$\psi = 9$		BUFLP	MILP	2DKP	AP	BUFLP	MILP	2DKP	AP
$p = 1$	$4 + 6(\psi - 2)$	22	46	18	20	22	22	40	42	44	44
$p = 2$	$4 + 6(\psi - 2)$	22	46	20	20	22	22	40	44	44	44
$p = \infty$	$4 + 10(\psi - 2)$	34	74	22	20	24	24	44	46	46	48
	Weighted sum			7				11			
	Other methods			10				18			

Table 2: For each method the worst case number of calls to CPLEX necessary to generate ψ points.

8 Conclusion

In this paper we consider the representation problem in biobjective mixed integer programming, and we used a desired cardinality of the representation as a stopping criterion as opposed to a desired quality level. We consider two measures that have become standard in the literature on representations, namely the coverage error and the uniformity level, and we show that the optimization problems of minimizing the coverage error and maximizing the uniformity level can be seen as duals of one another. By solving both problems, an optimality gap is therefore obtained, and in particular we show that this gap is closed if an equidistant representation can be found. Inspired by this result, we develop a method that attempts to construct an equidistant representation of a given cardinality by utilizing the space division technique of Voronoi diagrams. We show on a set of BOMIP benchmark instances that this method significantly outperforms methods from the literature both in terms of coverage and uniformity.

References

- T. Achterberg and R. Wunderling. Mixed integer programming: Analyzing 12 years of progress. In M. Jünger and G. Reinelt, editors, *Facets of Combinatorial Optimization*, pages 449–481. Springer-Verlag Berlin Heidelberg, 2013.
- Y. P. Aneja and K. P. K. Nair. Bicriteria transportation problem. *Management Science*, 25(1): 73–78, 1979.
- N. Boland, H. Charkhgard, and M. Savelsbergh. A criterion space search algorithm for biobjective integer programming: The balanced box method. *INFORMS Journal on Computing*, 27(4):735–754, 2015a.
- N. Boland, H. Charkhgard, and M. Savelsbergh. A criterion space search algorithm for biobjective mixed integer programming: The triangle splitting method. *INFORMS Journal on Computing*, 27(4):597–618, 2015b.

- N. Boland, H. Charkhgard, and M. Savelsbergh. The L -shape search method for triobjective integer programming. *Mathematical Programming Computation*, 8(2):217–251, 2016a.
- N. Boland, H. Charkhgard, and M. Savelsbergh. The quadrant shrinking method: A simple and efficient algorithm for solving tri-objective integer programs. *European Journal of Operational Research*, pages 1–13, 2016b.
- G. P. Cachon. Retail store density and the cost of greenhouse gas emissions. *Management Science*, 60(8):1907–1925, 2014.
- M. de Berg, M. van Kreveld, M. Overmars, and O. C. Schwarzkopf. Voronoi diagrams. In *Computational Geometry: Algorithms and Applications*, pages 147–163. Springer, Berlin, Heidelberg, 2000.
- M. Ehrgott. *Multicriteria Optimization*. Springer Berlin, 2nd edition, 2005.
- M. Ehrgott. A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research*, 147(1):343–360, 2006.
- M. Ehrgott, X. Gandibleux, and A. Przybylski. Exact methods for multi-objective combinatorial optimisation. In S. Greco, M. Ehrgott, and R. J. Figueira, editors, *Multiple Criteria Decision Analysis, International Series in Operations Research & Management Science 233*, pages 817–850. Springer, New York, NY, 2016.
- A. Eusébio, J. R. Figueira, and M. Ehrgott. On finding representative non-dominated points for bi-objective integer network flow problems. *Computers and Operations Research*, 48:1–10, 2014.
- S. L. Faulkenberg and M. M. Wiecek. On the quality of discrete representations in multiple objective programming. *Optimization and Engineering*, 11(3):423–440, 2010.
- S. L. Faulkenberg and M. M. Wiecek. Generating equidistant representations in biobjective programming. *Computational Optimization and Applications*, 51(3):1173–1210, 2012.
- H. W. Hamacher, C. R. Pedersen, and S. Ruzika. Finding representative systems for discrete bicriterion optimization problems. *Operations Research Letters*, 35(3):336–344, 2007.
- B. J. Hancock and C. A. Mattson. The smart normal constraint method for directly generating a smart pareto set. *Structural and Multidisciplinary Optimization*, 48(4):763–775, 2013.
- T. Kuhn and S. Ruzika. A coverage-based box-algorithm to compute a representation for optimization problems with three objective functions. *Journal of Global Optimization*, pages 1–20, 2016.
- M. Masin and Y. Bukchin. Diversity maximization approach for multiobjective optimization. *Operations Research*, 56(2):411–424, 2008.
- G. Mavrotas and D. Diakoulaki. A branch and bound algorithm for mixed zero-one multiple objective linear programming. *European Journal of Operational Research*, 107(3):530–541, 1998. ISSN 03772217.
- A. G. N. Novaes, J. E. Souza de Cursi, A. C. L. da Silva, and J. C. Souza. Solving continuous location-districting problems with voronoi diagrams. *Computers and Operations Research*, 36(1):40–59, 2009.
- A. Okabe and A. Suzuki. Locational optimization problems solved through voronoi diagrams. *European Journal of Operational Research*, 98(3):445–456, 1997.

- S. Saym. Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming. *Mathematical Programming*, 87(3):543–560, 2000.
- L. Shao and M. Ehrgott. Discrete representation of non-dominated sets in multi-objective linear programming. *European Journal of Operational Research*, 255(3):687–698, 2016.
- R. E. Steuer and F. W. Harris. Intra-set point generation and filtering in decision and criterion space. *Computers and Operations Research*, 7(1-2):41–53, 1980.
- T. Stidsen, K. A. Andersen, and B. Dammann. A branch and bound algorithm for a class of biobjective mixed integer programs. *Management Science*, 60(4):1009–1032, 2014.
- J. Sylva and A. Crema. A method for finding well-dispersed subsets of non-dominated vectors for multiple objective mixed integer linear programs. *European Journal of Operational Research*, 180:1011–1027, 2007.
- D. Vaz, L. Paquete, C. M. Fonseca, K. Klamroth, and M. Stiglmayr. Representation of the non-dominated set in biobjective discrete optimization. *Computers & Operations Research*, 63:172–186, 2015.

Proof of Theorem 1

Suppose that a representation $\tilde{R} = \{\tilde{y}^1, \dots, \tilde{y}^{|R|}\}$ exists such that $\delta_p(\tilde{R}) \geq \delta_p(R)$. Since R is an equidistant representation, $d_p(\tilde{y}^k, \tilde{y}^{k+1}) \geq d_p(y^j, y^{j+1})$ for all k and j . From this and Lemma 1 it follows that, if $y_1^j \leq \tilde{y}_1^j$, then $y_1^{j+1} \leq \tilde{y}_1^{j+1}$ for all $j = 1, \dots, |R| - 1$. From Lemma 3, $\hat{y}^1 = y^1 = \tilde{y}^1$, and therefore by mathematical induction $y_1^{|R|} \leq \tilde{y}_1^{|R|}$. However, by Lemma 3 $\hat{y}^2 = y^{|R|} = \tilde{y}^{|R|}$, and so $\tilde{R} = R$. ■

Proof of Theorem 2

Since any nondominated point y such that $y_1^j \leq y_1 \leq y_1^{j+1}$ for some $j \in \{1, \dots, |R| - 1\}$ is closest to (i.e. covered by) either y^j or y^{j+1} according to Lemma 1, it follows that

$$\begin{aligned} \gamma_p(R) &= \max_{y \in \mathcal{N}} \min_{y' \in R} d_p(y, y') \\ &= \max_{j \in \{1, \dots, |R| - 1\}} \max_{y \in Y(y^j, y^{j+1})} \min[d_p(y^j, y'), d_p(y', y^{j+1})] \end{aligned}$$

for any representation R . Given a representation $R = \{y^1, \dots, y^{|R|}\}$, let

$$\bar{y}^j = \operatorname{argmax}_{y \in Y(y^j, y^{j+1})} \min[d_p(y^j, y'), d_p(y', y^{j+1})]$$

for $j \in \{0, \dots, |R|\}$, where we follow the convention that $d_p(y^0, y') = d_p(y', y^{|R+1|}) = \infty$.

Given a complete equidistant representation $R' = \{y'^1, \dots, y'^{|R'|}\}$, let $R = \{y^1, \dots, y^{|R|}\}$ be a representation such that $y^j = y'^{2j}$ for all $j \in \{1, \dots, |R|\}$. Since y'^{2j+1} is the same distance from y'^{2j} and y'^{2j+2} , it follows that $\bar{y}^j = y'^{2j+1}$ for all $j = 1, \dots, |R|$, while $\bar{y}^0 = y'^1$. Moreover, $\gamma_p(R) = d_{R'}$.

Suppose that a representation $\tilde{R} = \{\tilde{y}^1, \dots, \tilde{y}^{|R|}\}$ exists such that $\gamma_p(\tilde{R}) \leq \gamma_p(R) = d_{R'}$. We first show that if $\tilde{y}_1^j \leq y_1^j$, then $\tilde{y}_1^{j+1} \leq y_1^{j+1}$ for any $j \in \{1, \dots, |R| - 1\}$. This is immediately true if $\bar{y}_1^j \geq \tilde{y}_1^{j+1}$, so we consider the case where $\bar{y}_1^j \leq \tilde{y}_1^{j+1}$.

If $\tilde{y}_1^j \leq y_1^j$, then by Lemma 1

$$d(\tilde{y}^j, \bar{y}^j) \geq d(y^j, \bar{y}^j) = d_{R'} \geq \gamma_p(\tilde{R}),$$

and so $d(\bar{y}^j, \tilde{y}^{j+1}) \leq d(\tilde{y}^j, \bar{y}^j)$ from the definition of $\gamma_p(\tilde{R})$. Since $\bar{y}_1^j \leq \tilde{y}_1^{j+1}$, it follows that \bar{y}^j is closest to the point \tilde{y}^{j+1} among all the points in \tilde{R} . Thus

$$d(\bar{y}^j, \tilde{y}^{j+1}) \leq \gamma_p(\tilde{R}) \leq d_{R'} = d(\bar{y}^j, y^{j+1})$$

and the desired result $\tilde{y}_1^{j+1} \leq y_1^{j+1}$ follows.

Since R' is a complete equidistant representation, $y'^1 = \hat{y}^1$, while by Lemma 2 $\tilde{y}^1 \neq \hat{y}^1$. The point y'^1 is therefore not in \tilde{R} and is closest to \tilde{y}^1 among all the points in \tilde{R} . Since

$$d(y'^1, \tilde{y}^1) \leq \gamma_p(\tilde{R}) \leq \gamma_p(R) = d(y'^1, y^1),$$

it follows that $\tilde{y}_1^1 \leq y_1^1$. Thus, by mathematical induction $\tilde{y}_1^{|R|} \leq y_1^{|R|}$. However, by Lemma 2 $\hat{y}^2 = y'^{|R|}$, and this point is therefore closest to $\tilde{y}^{|R|}$ among all the points in \tilde{R} . By Lemma 1 and the fact that $\tilde{y}_1^{|R|} \leq y_1^{|R|}$, we have

$$\gamma_p(\tilde{R}) \geq d(y'^{|R|}, \tilde{y}^{|R|}) \geq d(y'^{|R|}, y^{|R|}) = \gamma_p(R),$$

contradicting the fact that $\gamma_p(\tilde{R}) \leq \gamma_p(R)$. ■

Proof of Lemma 4

(a) Assume, to the contrary, that both y^L and y^R are dominated. First of all, note that they cannot be dominated by points outside $Y(y', y'')$, since otherwise these points would also dominate either y' or y'' , or both. Therefore, they each respectively have to be dominated by a point in the region of the other, i.e. y^L is dominated by a point $\bar{y}^L \in V_p^{\geq}(y', y'')$ and y^R by a point $\bar{y}^R \in V_p^{\leq}(y', y'')$. Since y^L is the right-most (locally) nondominated point in the region $V_p^{\leq}(y', y'')$, it follows that $\bar{y}_1^R \leq y_1^L$ and therefore that $y_2^L \leq \bar{y}_2^R \leq y_2^R$. Similarly, since y^R is the left-most (locally) nondominated point in the region $V_p^{\geq}(y', y'')$, it follows that $y_1^R \leq \bar{y}_1^L \leq y_1^L$. We have therefore shown that $y_2^R \geq y_2^L$ and $y_1^R \leq y_1^L$, but by Lemma 1 this implies that y^R is closer to y' than y^L , contradicting the definitions of y^L and y^R . The initial assumption therefore has to be false, and at least one of y^L and y^R is nondominated.

(b) Let y^L be (globally) nondominated and y^R be dominated by some point $\bar{y}^R \in V_p^{\leq}(y', y'')$. Above we have shown that in this case $y_2^L \leq \bar{y}_2^R \leq y_2^R$. Now it should also follow that $y_1^L \leq y_1^R$ such that y^L dominates y^R , since otherwise y^L would be closer to y'' than y^R . Similarly, it can be shown that if y^R is nondominated and y^L not, then y^R dominates y^L . ■